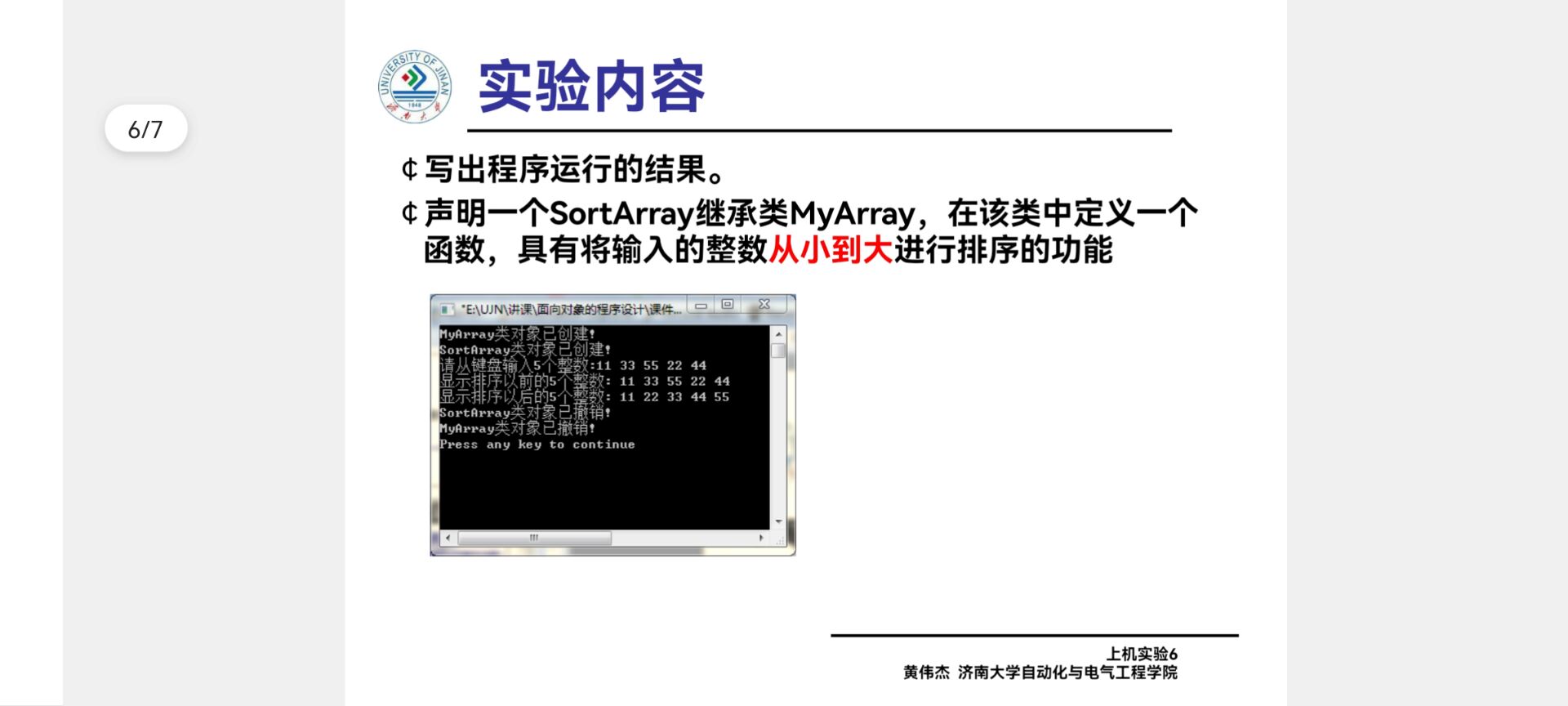
C++上机实验6



#include<iostream>

#include<string>

using namespace std;

class MyArray {

public:

MyArray(int length);

~MyArray();

void input();

void Display(string);

protected:

int\* alist;

int length;

};

MyArray::MyArray(int leng)

{

if (leng <= 0)

{

cout << "error length";

exit(1);

}

length = leng;

alist = new int[length];

if (alist == NULL)

{

cout << "assign failure";

exit(1);

}

cout << "MyArray类对象已创建！" << endl;

}

MyArray::~MyArray()

{

delete[]alist;

cout << "MyArray类对象已撤销！" << endl;

}

void MyArray::Display(string str)

{

int i;

int\* p = alist;

cout << str << length << "个整数:";

for(i = 0;i<length;i++,p++)

{

cout << \*p << " ";

}

cout << endl;

}

void MyArray::input()

{

cout << "请从键盘上输入" << length << "个整数:";

int i;

int\* p = alist;

for (i = 0;i < length;i++, p++)

cin >> \*p;

}

class SortArray: public MyArray {

public:

void Sort();

SortArray(int leng):MyArray(leng)

{

cout << "SortArray类对象已创建！" << endl;

}

~SortArray();

};

SortArray::~SortArray()

{

cout << "SortArray类对象已撤销！" << endl;

}

void SortArray::Sort()

{

int i, j, temp;

for(i = 0;i<length-1-i;i++)

for (j = 0;j < length - i - 1;j++)

{

if (alist[j] > alist[j + 1])

{

temp = alist[j];

alist[j] = alist[j + 1];

alist[j + 1] = temp;

}

}

}

int main()

{

SortArray s(5);

s.input();

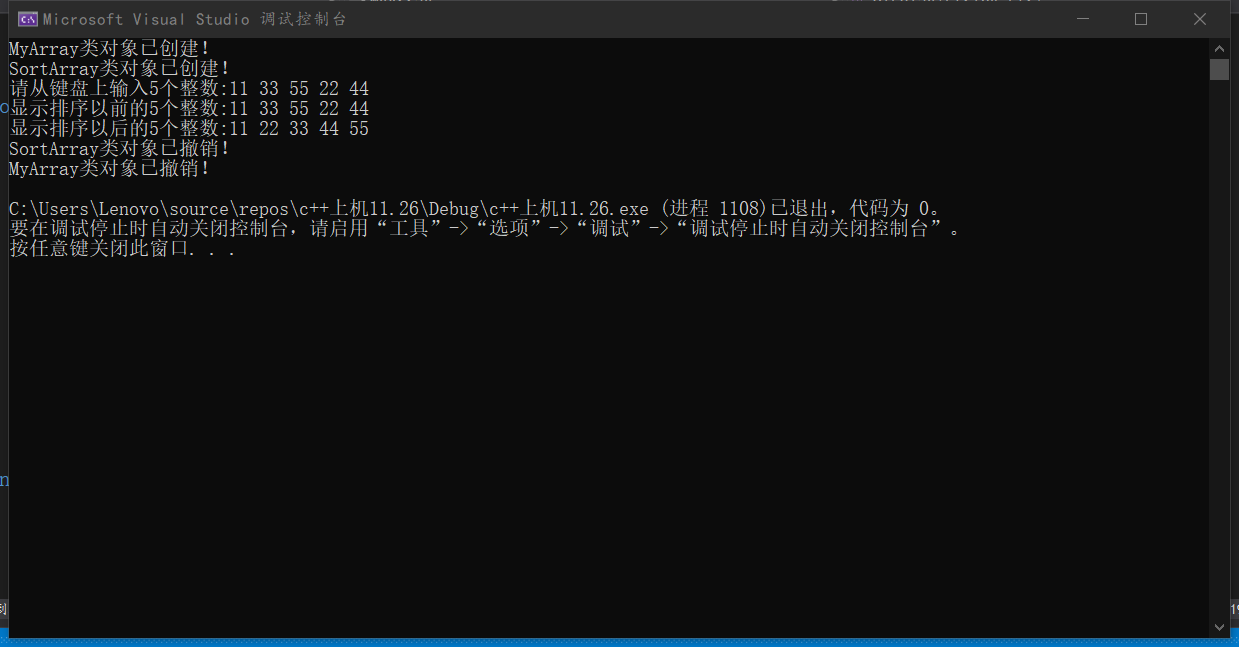
s.Display("显示排序以前的5个整数：");

s.Sort();

s.Display("显示排序以后的5个整数：");

return 0;

}



**心得**

第六次实验的目的主要是了解派生类的继承方式以及派生类的析构函数的使用方式。了解了继承分为多种，公有继承，保护继承，私有继承。而且在这次老师说讲的课件中，老师让我们定义一个函数对数据进行大小排列，就体现了继承和析构函数的功能和特点。

Copyright 2021-2099 Kongyouhao. All rights reserved